ACE

Atari
Computer
Enthusiasts

3662 Vine Maple Dr. Eugene OR 97405

OCTOBER, 1986
Mike Dunn, Jim Bumpas, Larry Gold, co-editors

News and Reviews

by Mike Dunn

This will be the first article that I have written with my ST, and also with **Thunder**, the on-line spelling checker by Batteries Included (reviewed last issue). When I arrived home after my trip to England, I found a number of letters inviting me to visit from our members in the area. I am sorry I did not visit those who invited me, but the letters did not arrive until after I had left.

We are still working out the bugs for the bulk mailing; the post-office has just changed all the rules again. Our issues can now be larger, so hopefully the trade off will be worth it. Please let us know if you do have problems. Our BBS has been down for a variety of reasons, including the power line being hit by lighting, but is back to normal now.

For those of you with 130XE, we now have **TurboBASIC** from Holland thanks to Bob Cook. An article from the Western New York Atari Group explains how to use it. This is the fabulous, very powerful BASIC which uses the extra memory of the XE and is very fast.

Remember Rita Plukss Australian series of articles on Mandelbrots, fractals, etc? She has sent us much more on fractals in both 8 and 16 bit BASICs as well as LOGO, so starting with this issue, you will learn alot about this fascinating subject and also how the old and new Atari computer BASICs differ.

The Portland Atari User Group and Atari are sponsoring a Northwest Atari Show in October — see article in this issue.

We recieved many programs for review this month, and, as is our custom, they have been given to a variety of members to review. (We like to favor those who help with the newsletter, including especially those who attend our collating parties — ed.) There are a number of new reviewers this issue and I hope you enjoy them. We also welcome reviews on hardware or software you buy.

Abacus (POB 7219, Grand Rapids, MI 49510) has released their new book, **Atari ST LOGO User's Guide**. I have not done much with LOGO in the past, but this book is very good. Filled with short programs illustrating the Keyword explained, it is a lot of fun to see how easily it is to do complex graphic figures. It covers many things beside graphics, such as list processing, procedures, recursion, and other advanced programming concepts. Highly recommended. They also sent us their new wordprocessor, **Textpro**, reviewed elsewhere in this issue.

Xlent Software (POB 5228, Springfield, Va 22150) has released their new ST version of one of their most popular programs, **Megafont-ST** . Xlent is known for their very powerful, but not always easy to use graphic programs. This program is the first very easy to use program from them. Using the usual GEM interface, with pull-down menus, etc., you can use this printer utility with 1st Word, D.E.G.A.S., or ASCII files, and works with Epson, Gemini, and Prowriter type printers. Using your mouse, you can select 8x8 or 8x16 matrix fonts, in small, medium, large or extra large sizes in about 26 different styles. There is also a built-in Font Editor, as well as formatting commands to print the page as you want. You can also mix text with graphics from D.E.G.A.S. For all you fans of Xlent products, you will be amazed how easy it is to do all of the above! Only $40. Xlent have also announced their new program, **Typesetter Elite** , which allows a full page layout, what-you-see-is-what-you-get page. It loads D.E.G.A.S. and ASCII files, and includes D.E.G.A.S.-like drawing tools. If it is as easy to use as Megafont-ST above, they will have a real winner in the hot desk-top publishing market and give buyers of the upcoming Paper-Clip Elite a real choice. We will have an extensive review of both if and when we get them.

We have received several very interesting ST demo disks. Print-Technik (Nikolaistrasse 2/368197, 8000 Munich 40 W. Germany) sent a disk of color pictures from their video digitizer, and another disk requiring 1040K of pictures which talk with fabulous digitized sound!. Stone Age Software (POB 1216, Amherst, NH 03031) sent two "PartialWare" programs. These allow you to do most of what the commercial version does, but are somewhat limitied. For many people, these versions will satisfy their needs; the more complete versions cost money. Lewis123 is a spreadsheet for chemistry students, etc., dealing with covalent bonds and organic chemical bonds ($30) and Encrypt puts your data in code' at 1000 bytes for second ($20). Contact Jim Bumpas for your copy of any of the above.

Magafont-ST print styles

This is ARCHAIC font.
This is BIGBLUE font.
This is BIGSTUFF font.
This is COMPUTER font.
This is CURSIVE1 font.
This is CURSIVE2 font.
This is FANCY1 font.
This is FANCY2 font.
This is FANCY3 font.
This is FINE font.
Thιω ιω ΓΡΕΕΚ ΦΟΝΤ.
This is ITALICS font.
THIS IS MASH FONT.
This is OLDE font.
This is OUTLINE font.
This is ROMAN font.
This is SPECIAL font.
This is STANDARD font.
This is SCOP font.
This is ZEBRA font.

This is ARCHAIC font.
This is ASCII font.
This is CLASSIC font.
This is COMPUTER font
This is CURSIVE font.
This is DAISYW font.
This is FANCY3 font.
This is FANCY4 font.
This is OLDE font.
This is RALLY font.
This is SHADOW font.
This is STANDARD font.
This is STENCIL font.
This is THIN font.
This is THIN2 font.
This is WOODCUT font.

the new XE/XL 8-bit 64K public domain BASIC from Holland & Germany.

Fully Atari BASIC compatible, includes a COMPILER & full instructions and demos.

Thanks to the Western NY Atari Group- other groups please obtain from them directly.

Articles continue in next ACE. Requires at least 64K. Very fast and powerful; many new commands. A must for all BASIC users!

turboBASIC
$10

BUMPAS REVIEWS

I spent the Labor Day weekend in the San Francisco bay area playing in a Colonial Conquest tournament at the Pacificon game convention. SSI sponsored the event and provided $180 worth of prize certificates. I helped organize the tournament and won 2d place (I used my certificate to get a copy of **Rails West** . For those of you who don't recall, **Colonial Conquest** is a game of diplomacy for up to 6 players (UK, France, Germany, Japan, U.S., and Russia) during the colonial era.

Even though the graphics on this 8-bit game are blockish and rather crude, amazed on-lookers commented on how much better the game looks than on C-64 or Apple computers. And SSI reports they are making a conversion for the ST machine. The tournament was so successful, I hope to do it again next year. If you want to know more about Pacificon, write them at: Box 5548, San Jose, CA 95150.

While I was there, I also went to SSI's new offices in Mountain View and was given a tour through their R & D department. They've got a C-Amiga and Atari ST side-by-side. C-64s and 8-bit Ataris are also side-by-side. I saw an IBM PCjr. And of course, plenty of Apples. They informed me they've dropped the project of converting their **War in the Pacific** to the Atari. Two reasons are given: One, the Atari disks don't hold enough data at 90k. The second reason is more general and applies to all software: Atari users don't buy as much software as other computer users. They explain to me that the installed base of Atari computers is twice that of Apple, yet more Apple software is sold than Atari software. The installed base of C-64s is twice the size of Atari, but C-64 users buy more software than is reflected merely by the larger number of machines. SSI intends to continue supporting the 8-bit Ataris, but they're not so eager as once they might have been to convert all their titles to Atari.

If computer users (of any brand) want to insure a continued supply of new and innovative software which is of good quality, then they need to provide a market in which software developers can prosper. If not, the software talent will go to those markets where they can prosper. This is especially critical now for the 8-bit machines. Commodore has tried several times to kill the C-64 line but had to re-start production because of demand. The 16-bit machines threaten to push the 8-bit machines off the market. And it will happen, sooner or later. But it can be later if the software market survives. Apple seems to be taking the route of up-grading their 8-bit line into the 16-bit world. Atari could market a Charlie Andrews-type mega-memory 8-bit machine with nearly the capabilities of 16 bit machines. This kind of activity could extend the life of the 8-bit market. If Apple and Atari had done this 2 years ago, the 16-bit market might have been still-born. But no machine will survive in the market without demand for software.

Of course, our 8-bit Ataris will still do everything they've been doing for the past several years. And we do have some top productivity software — word processors, file managers, spreadsheets. And games. But the new software will do all these things better and more enjoyably. Some of that software will be for the 8-bit market if there is a demand.

FLASH

Antic has released an updated version of their excellent terminal program, **FLASH**. This new version corrects some of the problems experienced with the VT100 emulation and adds some new features. You can now CTRL-W to ''wipe'' the capture buffer from the terminal keyboard. You can also set the baud rates for the remote systems to which you connect right on your dialing directory. In this way, the baud rate for the system you call is automatically set upon dialing the phone. Flash also supports the hi-res mode by offering the option of a 48-line display.

The VT100 emulation is the nicest thing (for me) since sliced bread. I've quit using PC Intercomm in favor of Flash. I have a 1-meg 520 ST running as a smart terminal on an Alpha-Micro 1072 (which only had 894k RAM when I started working here). With Flash, I can run at 19,200 bps which is faster than the 15 dumb terminals on the system (they're at 9600 bps). With PC Intercomm, I could only run at 9600 bps, too. Its screen handling was written for the IBM. It doesn't seem to be able to handle the speed of the 68k chip. At 19200 bps, PC Intercomm dissolves the screen into unreadable garbage. **FLASH** literally ''splashes'' the screen on the CRT. You can hardly detect cursor movement when it paints a screen. There is only one thing I was able to do with PC Intercomm which I cannot with Flash. SuperVue, the Alpha Micro word processor, shows on-screen underlining with PC Intercomm. It does not with Flash.

FLASH is the best terminal program I've seen on ANY micro (and I've used SmartComm, Crosstalk XV, PC-Talk, and others on the IBM). At under $40, it's far and away the best value in modem-land.

GETTYSBURG

Gettysburg: The Turning Point ($60, SSI) refines the game system introduced with **Antietam** and simulates what is probably the most important battle in the history of the North American continent. Players can concentrate upon strategy even more than in the Antietam game. The laminate conference map now marks the terrain features which contribute to victory. The hourly status reports now show complete records of activity which contributes to victory: Charts show casualties of infantry, cavalry, guns, officers, etc. A page describes the attitude of corps and division commanders as ''confident'', ''confused'', ''cautious'', etc.

Players of Antietam learned to do a few things which are not possible in real life — or in the **Gettysburg** game. If you can force a unit to retreat in Antietam, and the unit is surrounded by stacks of two friendly units with no empty hexes, the unit will be destroyed just as if it were surrounded by enemy units. In **Gettysburg** this has been changed to permit movement through stacks of friendly units by routed units and units in column. Another thing players used to be able to do is to expend operational points to explore nearby terrain to look for enemy units. If you don't want to move there, you just press the ''z'' key and the move is erased, regaining all the expended ''OP''s. You can still do this, but each time you do, it costs you several OP points.

Other changes include a completely re-vamped command control system. Ammunition is now a finite supply for both small arms and artillery. The map shows dots in the center of each hex for ease in finding ranges. Reinforcements include the option of a variable entry time. Battle-field smoke can reduce visibility so much that units will not see any target to shoot. Units can be ordered to shoot at a hex not currently occupied by a target. They can also be set not to shoot at all. And units can be set not to receive any ammo re-supply (why might one want to do this? If the unit is so small, or of such low morale that it cannot hit anything anyway, why re-supply it?)

Antietam has been a top-selling piece of software for SSI. **Gettysburg** promises to outsell Antietam. If you enjoy ''strategic simulations'' as much as I do, you'll want to have this software.

CALENDAR

Calendar (MichTron, $30) is the best desktop appointment and alarm calendar I've yet seen for a microcomputer. The program runs as an accessory under the GEM system on Atari STs and IBM PC/XT and compatibles.

Windows open into the calendar like boxes inside of boxes from the ''Year'' screen, to the ''Month'' screen, to the ''Day'' screen. You can set entries on the Day screen for every 15 minutes in the entire 24-hour day. An alarm can be set for any entry, and the system time can be set to ''chime'' every hour. The chimes and alarms will only interrupt a program running under the GEM desktop. For other programs, the dialog boxes will wait until you return to the desktop. There is also a ''direct entry'' mode with which calendar entries may be inserted without stepping through the calendar windows to the day desired.

If you live past the year 2099, you'll need another program. And while you may insert as many messages or alarms as desired, the program will only retain the most recent 999 entries. The program takes about 50k of RAM. One enhancement I want to see is the ability to set automatic recurrences of the alarms without having to make separate entries (i.e., annual birthdays, or perhaps wedding anniversaries of several persons) and to plot blocks of time when selected persons are available or unavailable will be nice, too. But this program does all it promises and at a reasonable price.

There is a page containing the most intelligent (and **accurate**) description of the purchaser's rights and the rights of the copyright holder I've ever seen with a piece of software. It makes clear the copy of **Calendar** is your property when you buy it. You may sell it in its entirety (and you must also include in the sale all backup copies you've made, or you may destroy the backups). You may use your copy on as many computers as you own or have access to, at home or at work. On a single computer, you may permit the program to be used by as many people as have access to that computer. All this without violating the copyright. Multicomputer systems and Local Area Networks may apply for quantity discounts and site licensing agreements, as they are excluded from the above uses.

We've also received v.1.0 of **Universe II** from Omnitrend. It's a massive game, so we can't have a review before next issue.

TEXTPRO

TextPro ($50, Abacus), the complete word processor for the Atari ST. Or at least this is the idea you get from the Abacus Software advertisements and the description of the program on the box slipsleeve. Abacus Software, has released a GEM based word processor, written by the Data Becker Group, which could make writing easier for some ST users.

ST TextPro is a totally new program not just a port of some other product. According to the manual, it was written by professional writers, for professional writers. All three authors had used several word processing packages before but none of them measured up in terms of speed and features, hence TextPro was born. So then with this in mind let's take a little tour through the features of this new entry in the ST's growing line of word processors.

There several philosophies to word processing. Under one philosophy, text is displayed exactly the same way as it is printed on the paper, ala 1st Word. Under the second, text is entered and displayed as a continuous stream of characters. The word processor pays no attention to either line or page arrangement. And text is arranged for printing by using format commands. This is the approach the writers of TextPro took.

The first thing you should know about TextPro is it is a copy protected program (please send your letters to Abacus regarding this). You can copy the files to a hard disk, but the original must be in drive A: in order for the program to function properly. Next you will notice TextPro is not just one program, but a collection of programs, there is TEXTPRO.PRG, the word processor; OUTPUT.PRG, a print formatter/mail merge program; CONV.TTP will convert a 1st Word file to a TextPro file; and SPLIT.TTP to take large files and cut them in half.

The next item you must know is you MUST read the manual. TextPro is not a program you can take straight out of the box and use, it has many advanced features and complicated formatting sequences. To help you there are 30 programmable function keys, each capable of holding up to 160 characters. And most all commands have a key combination alternative to the mouse and menus. I will try to focus this review on what features are there instead of how you implement them.

The familiar menu line greets you once you've loaded TextPro and the menu headings are: Desk, File, Edit, Style, and Formats. In the Desk menu you have the usual desk accessories and program info. The info screen shows how much memory you've used and how much you have left. Then the File menu contains load and save commands along with a delete and drive select choices, you can choose any drive from A: to E: or even the RS-232 to receive text from a modem. Here too you have a choice of which type of file you want to save: A DOC file includes all formatting commands, tab settings, header and footer information, etc., a NON-DOC file saves text as a straight ASCII file and a C-SOURCE file which automatically justifies braces ({ or }) and does auto indentation. You may also output the file directly from TextPro to the screen or a printer. Lastly, function key definitions are loaded, saved, and altered from this menu.

The Edit menu has a host of features making this word processor unique among those currently available for the ST. Automatic hyphenation, automatic indexing, automatic table of contents generation (complete with page numbers), a place holding feature to perform merges during printing, along with search and replace functions. You can move to any line of text in your document with the Goto line . . . command, or insert any number of blank lines or page breaks. There are also block copies, moves and erasures. And a sort function for indexed words. All these commands to make writing easier for the professional writer.

The Style menu is similar to the one available in 1st Word, but there are some differences. Many printers allow character spacing other than 10 cpi pica, and TextPro allows 12 cpi and 15 cpi. It will also change the character under the cursor from upper to lowercase and back. One last feature of this menu are the Time and Date functions, so if you have a battery backed up clock (TimeSaver, Logikhron, etc.) the current time or date will be inserted at the time of printing.

The Formatting options include the ability to indent to the left, or right, used to set off paragraphs. Left, Right, or full justification, and the option of centering text. Tabs can be set anywhere along the line and there are two text entering modes, typeover and insert. Under the formatting commands you can change line spacing, lines per page headers, footers, columns per page, column width, along with top and bottom margins.

There are also several ways to move around in your document, besides the mouse & button, the cursor keys, both alone and in conjunction with the control and alternate keys help speed you through your document. The backspace and delete keys also have similar options.

Now we get to the part where TextPro is set apart from the others, print formatting. The OUTPUT program does all the table of contents printing, index printing, etc. It will print your file on a printer or to a file, or even to a design file to be used with the soon to be released Page Designer to incorporate graphics with your text. And if you own an Epson FX or compatible printer you also have the option of printing your text sideways. You can customize the printer driver to include other printers than the ones included, although this is no easy task. All in all, the program does perform as advertised and may make casual writers become more professional in their writing.

Authors note: There were some minor bugs in the version I received, please test all desk accessories you use with TextPro before you buy this program, as some will not function properly. THUNDER! does not seem to work interactively with TextPro. TextPro will probably be easier to learn for those moving up from ST-Writer as they both use a similar design philosophy and TextPro documents can be directly loaded into ST-Writer with no conversion. Now I'm waiting for Microsoft Write.

— Buddy Hammerton

TEN MILLION SHOTS

There is a new joystick available for the Atari computers, the EPYX 500XJ from EPYX, Inc. of Redwood City, Ca. It will also work on the Atari 2600 and 7800 series as well as (I have to say it) the Commodore computers. This joystick differs from any other I've used in more than one way. The most obvious difference is the shape. It is asymmetrical with a rounded bottom. This joystick is not meant to be placed on a desk or table, instead, it's designed to fit in your hand. The contacts are microswitches which give a very positive feel as well as an audible click. The result is much more precision than with joysticks that use contact switches. Another difference is the warranty that comes with this joystick: Five year, 10 million shots. I'd be willing to keep track of the time but, I'm assuming that EPYX will take your word on the number of shots. Anyway, with a five year warranty, this joystick will probably outlast your computer.

How good is it? The feel is positive, the switches give excellent feedback and I doubt you will ever break the steel shaft on the stick. However, I personally can't use it for more than a few minutes because it feels too big for my hand. My hand starts to ache after any intense playing. I love this stick for precision drawing or pointing but not for high intensity games.

This joystick is designed for right handed users so it may be a problem for some of you. Since comfort is purely subjective, if you're looking for a high quality joystick, you should give this one a try. It's an excellent choice if it is comfortable for you.

— Steve Golden

TIME-SAVER

(UPDATE)

If you've been trying to purchase TIME-SAVER, the battery powered internal clock for the ST, you may have run into some trouble since the distributor, MIND-MINE, has gone out of business. Fortunately, TIME-SAVER is now available directly from the designer, UNILAB, 11721 Roosevelt Way N.E., Seattle, WA 98125.

As we go to press... I just received a notice that MIND MINE is back in business at a new address. MIND MINE COMPUTER CENTER, 1964 130th AVE. N.E., BELLEVUE, WA 98005. For those of you who have MIND-MINE'S 1-Meg upgrade, I hope it continues to work as well as mine, but it is comforting to know MIND MINE is back (just in case). Also, I'm looking forward to more fine Atari products from them.

— Steve Golden

WALDEN ASSEMBLY

The Alcyon C compiler comes with a 89K+ runtime file containing numerous routines all written in C. C is quite efficient as high level languages go, but it is no match for assembly in either size or speed. If you look at the benchmarks in the back of the July issue, you will see how much difference it makes to have the runtime written in assembly.

If there is enough interest, this will be the first in a series of articles allowing you to replace most or all of the runtime routines which come with the Alcyon C compiler with routines written in assembly. The resulting runtime will be about 1/10th of the size, and take 1/4th of the time to link with your regular programs. They will take up less space in your programs, and run faster.

With a minimum of effort, you can probably change them to work with any C compiler — you could even use the non i/o routines on other 68000 computers. It should also help to familiarize you with 68000 assembly language.

The first 4 fourtines are string handling functions. The structure of 680000 assembly language makes string handling extremely efficient and easy. The Alcyon C compiler passes arguments on the system stack. The first argument will be in the current stack plus 4 (the first 4 bytes store the return address). The next argument will be at stack plus 6 or 8 depending on whether the first argument is a word or long. A char placed on the stack will still take up 2 bytes in order to keep the stack at an even address. Registers D0-D2, and A0-A2 are "scratch" registers which we can use without saving the old values. The return value if any, is always placed in register D0.

— Ralph Walden

EPICYCLOIDS

OR WHEELS WITHIN WHEELS

by Andrew Plukss

EPICYCLOID: The path traced out by a point on the circumference of a secondary circle (the epicyclic circle) which rolls around a primary circle.

In the days before the price of microchips fell to the levels which made home computing affordable, I used to derive a great deal of pleasure in drawing patterns with a Spirograph set. It was really quite simple — you only had to place a pencil inside one of a number of small holes in a wheel which was then rotated inside (or around) a second fixed wheel thereby generating the pattern. The final shape of your creation was controlled by the relative sizes of these wheels. I could even tolerate the occasional disaster when an almost completed pattern was ruined by a pin holding the main wheel working itself loose at the most inopportune time. The Spirograph was fun and I am sure still provides a great deal of fun for many people today.

I don't seem to have progressed very far over the years with this simple pleasure except that the Spirograph set has been replaced by more advanced tools, firstly the old reliable, still going strong, Atari 800 and now the more sophisticated 520ST. (The Spirograph has suffered the ravages of time and lies in a dark cupboard at home, sadly missing a few wheels.) One of the problems for me in drawing these Spirograph-type patterns, or to give them their correct term epicycloidal patterns, has been in trying to work out the exact formula to generate a desired shape. The basic formula is not too difficult, being of the form $f(x) = f(a) + f(b) + f(c)$ where each of the terms on the right-hand side of the equation represents a sine or cosine function such as $f(a) = R\sin(Ua) + S\sin(Va) + T\sin(Wa) + \ldots$ etc. with the number of terms in the equation increasing with the complexity of the final pattern. The real difficulty lies in working out the values of the constants R,S,T,U,V,W etc. and until last year this provided a headache or two. My problem was solved with the release of a book by Ross Edwards (see illustration) which not only provided 643 examples of possible patterns but also gave the relevant coefficients in the basic formula to generate these designs. If you are interested in this form of screen graphics, then I can thoroughly recommend this book. It is quite fascinating in many ways. Not only do you learn how to apply modern computer methods to generate an infinite number of patterns but also gain some insight into the historical development of epicycloidal pattern generation as a decorative tool. Nineteenth century engravers popularised this art form through geometrical engraving on wood, metal, ivory, and particularly on glass. An interesting piece of trivia regarding Edwards' book is that although it is firmly aimed at today's home computer user, the most recent book mentioned in the bibliography was published in 1844!

I have included versions for both the 800/130XE and 520ST computers to generate these epicycloidal patterns as well as suggesting a few more avenues to explore. Both programs include screen save routines but a screen retrieval routine is only included in the 520ST version. 800/130XE owners can load saved screens via utilities such as Fader or Fader2 from Antic magazine. Humpty Dump from Kidstuff Software also has excellent utilities for this purpose with the added bonus of being able to print screens at any stage of their development. Make sure you use the extender .MIC for your saved screens if you anticipate using the Fader programs. The examples accompanying this article were generated on the 520ST. 800/130XE owners will find that these and some of the more involved examples from Edwards' book may require a resolution greater than that possible on their computers. There is really only one way to be sure though — try it! The speed of plotting will decrease markedly with the complexity of the pattern and with a decrease in step size. The running time of this program could be significantly improved by making use of the symmetrical nature of many patterns and the method for this is explained quite clearly by Edwards.

Earlier, I mentioned that epicycloidal patterns can be thought of in terms of a line being traced out by a point on a wheel rotating around a second wheel. The term R represents the radius of the second or fixed wheel, S is the radius of the rotating wheel. R+S thus is the radius of the curve or pattern generated. (Later another wheel and corresponding term T are added, increasing the pattern radius to R+S+T.) N is a factor that influences the final shape of the pattern. As the pattern takes shape you will notice that a series of loops or whorls are drawn around a basically circular shape. If N is positive then the loops are drawn inside the circle, if N is negative then the loops form outside the circle. (The number of loops drawn is equal to N-1 and N+1 respectively.) The input to both programs requires values for NA and NB where N=NA/NB and as in most cases NB=1 you may wonder why not simply use N and enter a fractional value when this is necessary. The reason is that if N is fractional then pattern generation is such that a point on the rolling wheel only returns to its original position after NB rotations and hence the iterative plotting routine must take this into account by increasing the number of iterations appropriately (line 300 in both programs).

A few patterns to try out:

vp's rambling

I hope all of you have filled in the questionnaire and sent it to the address listed. We need all the information it will provide so that we can do a better job with the newsletter and everything else we do for ACE.

Please check the dates on the address label to find out if you have to renew your subscription. Without your subscriptions ACE will cease to exist so please check and if you see that it is renewal time and send it in to us. We hope what we do pleases you for we cannot exist without your support.

Atari is going public and soon they will be selling stock in the company and if you believe in Atari you might think about buying some.

According to the ads I see in the magazines and newspapers there are some really good buys in software and hardware. This seems to be the time one should think about filling in the blanks on what you need to make your system as perfect and you can.

— Larry Gold

TRAP THE DRAGON

The object is to trap the dragon by moving block walls around. You surround him and keep pushing blocks in until you squeeze him to death. You can't get next to him though. Even one space diagonally and you've had it.Use a joystick in port 1 to move the walls. Pointing without the trigger pressed moves you (the blue square). Pressing the trigger while pointing the stick moves a block if you are next to it. You may find the easiest thing to do is to hold the stick in one direction and continually press and release the trigger.The higher the level of difficulty, the fewer the number of blocks you have to move. Happy dragon hunting!

— Stan Ockers

CORNERMAN

The hour is late, you are all alone, a faint glow fills the room. A thought comes to your mind and you reach over to jot it down on the cardboard used for your note pad. Next, a few quick equations on your calculator, and before you finish, the batteries go dead. Nothing seems to be going your way. Oh well, you can always give a call to the person who said he'd help you out in that time of need, but where did you put that phone book.

Suddenly a knock comes at the door. You ask yourself, 'Who could be at my door at this hour?!'. Slowly creeping to the door, you ask, 'Who is it?'. The answer is John Weaver, and after you open the door he hands you a box, then just as quickly as he appeared, he disappears.

You examine the box carefully. Noting the name, CORNERMAN, The Ultimate Desktop calculator, note pad, phone book, phone log, a window to DOS, printing utilities, and even a game. These are all rolled up in one neat little package, to be used anytime from the GEM desktop or from any GEM based programs.

Now you know what CORNERMAN can do for you. Instead of writing the usual 'rewrite the docs' review, here are some of the highlights of a recent phone conversation I had with the author, J. Weaver Jr.

You should all be aware this man is not in this business solely for the money. He has a genuine love of programming, and the programs he writes are things he wants to see. He started by writing software for CoCo's and then moved on to Sanyo MB-555's, and now he has the ST, which he calls, "the greatest personal computer." Many of the items we find in CORNERMAN are things he felt were necessary to make life with his PC more useful.

We also discussed some specifics about CORNERMAN I want to pass on to you. When you read through the manual one thing you will notice, is there is a phone number listing for Factory Programming. Believe it or not, this is a direct line to Mr. Weaver himself, as a matter of fact, no one else in the office is allowed to answer the phone. When he is there, he will take time out from his busy schedule to help you get the most out of CORNERMAN. Make sure though, to call after 2:00 PM eastern time. Or better yet, drop him a line through the mail, either direct to Factory Programming or Michtron. And for modem users, please feel free to join them on the Michtron RoundTable (RT) on the General Electric Network for Information Exchange (GEnie). He and Timothy Purves are the Sysops of the RT.

These people are very responsive to the end user, as we can see by the way they release enhancements and updates on a regular basis. There are patches already available for owners of Avatex modems, so they can use the phone dialer and also one to correct a minor bug in the calculator.

Which brings me to another point, the calculator. Calculators now are cheap, walk in to any K-Mart and you can pick one up for as little as 97 cents. So why might you ned an eight digit calculator on a computer? Well, that's what John thought, so he provided us with a sixteen digit one, that's right sixteen digits. What's more, it will do arithmetic on HEX values, OCTAL values, BINARY numbers, as well as DECIMAL.

Of course, we all know that a piece of software is never finished, and such is the case with CORNERMAN. Mr. Weaver is working on a way to speed up the square root function of the calculator, and a version of CORNERMAN for the IBM PC running GEM.

So then what else can we expect form Michtron and Factory Programming in the future, Laser printer drivers for the ST, GEM based Mi-Term for the IBM, a sort of CORNERMAN Construction Set. And I hope, lots more.

One last note about CORNERMAN, I like it. And to use John Weaver's own words, "I wrote it because I needed it," and "I use it every day."

— Buddy Hammerton

USER HINTS

From the September, 1986 issue of TACE: "The ST version of **PrintMaster** has the best graphics editor out of any of the other versions. One of the nicest features is the 'Use Window when Moving' option. However, it has never been fully documented (not even in the revised ST manual). The windowing feature allows you to cut a little part out of a graphic and drag it around to a new location, or rubber stamp it into many locations. I discovered you can also use it to cut and paste between different graphics!

For my tutorial, I'll use graphics located on the PrintMaster master disk, so everyone with the program can follow along. Ok, first load PrintMaster and go to the graphics editor.

1. Enter "L" to load a graphic. Choose the 'By Picture' option when asked to choose a graphic. Go through the pages of choices till you get to the '?' (question mark) graphic and choose it.

2. Now you are back in the graphic editor. Move the cursor so it is at the top left edge of the question mark. Press 'W' (window) and rope off the question mark as closely as possible with the 'rubber band'. Click the mouse button when you have finished. Now move the question mark as far as you can to the right edge and click the button to stamp the image in that position. Next press ESC to get rid of the window. Go clean up any residue left by the old question mark. After that, SAVE this modified graphic on a data disk using CTRL-S.

3. Follow the same procedure to load the '!' (exclamation point). Rope it off with the 'W' feature as closely as possible.

4. Now, without getting rid of the window, use the 'L' option to load the modified version of the question mark. Move the exclamation point into the blank space and click the mouse button to drop it there. Press ESC to get rid of the window and now you have a combined question mark and exclamation point.

Neat, hun? You can use this to integrate the best part of any PrintMaster graphics. It is not as flexible as Clipboard features and it doesn't make the white space invisible when pasting, but, with clever arrangement of the pieces and a little cleanup of the finished image, it can make creating and customizing your graphics a little bit easier.

— Donald Lusk

NULL JOYSTICK

(reprint: Mid Michigan Atari, August, 1986)

Many new owners of the Atari ST have probably come across the problem of transferring files from their old 8-bit Ataris. In most cases, getting the ST to talk to their modem was just a matter of purchasing or building a cable, acquiring a terminal package, and then calling up the local BBS. After the initial thrill of getting the ST up and running they probably started to notice there is still a lot of good software available for the 8-bit machine. All they have to do is download it . . . Well, maybe they will reconnect the 8-bit machine to the modem and call back. Of course this is also the time when they find that elusive ST subroutine they have been looking for so then they have to disconnect, reconnect the modem to the ST, re-call the BBS, then . . .

Of course, what about the disk subscription you have to Antic magazine? How the heck are you going to get those ST programs off that 5.25" disk? Do you really have to upload all those files to the local BBS with your 8-bit then call back with your ST and download them back onto the 16-bit machine? Sigh. If only you had bought the 850 interface. Etc., etc.

Well, here is an easier way to transfer files and you don't need an 850 interface; or two modems; or a friendly SYSOP in order to do it. The solution is using a null modem cable. The cable gives you a very bare-bones RS232 configuration of a signal ground, a receive data, and a transmit data line. In order to make a null modem all one has to do is connect the receive data line from the 8-bit to the transmit data line of the 16-bit. Likewise the transmit data line from the 8-bit is connected to the receive data line on the 16-bit. Signal ground is common to both.

After building the null modem I then used STTERM2 on my ST and Smartterm 5.0 on my 8-bit. I set both terminals to Xmodem protocol, 1200 bps, ASCII, and half duplex. I successfully transferred several test files from one Atari to the other without fail.

The cost for the cable will be all of about $13, and that is if you buy all your parts new from Radio Shack. If you have an old Atari joystick, you can use the cable from that and save even more.

Parts list: Atari joystick plug and cord — 2761538; length of 3 strand wire; switching diode — 2761122; 4.7k ½ watt resistor — 2718030; 25 pin RS232 female connector — 2761548.— Rick Beetham, National Capital Atari User's Group

FRACTALS

Exploring between dimensions — the outer/inner limits!

by Rita Plukss

The Mandelbrot set has been put away for a while, and I have once again returned to where it all started; FRACTALS. Dick Kellett and I have been working on fractals for some time now, and with the recent upsurge of popularity and interest in this area (especially on the ST) I decided to include some of our listings in the magazine.

A fractal is really quite beautiful as well as fascinating. It has a recursive property; the patterns tend to replicate themselves on a smaller and smaller scale. (For information regarding types of fractals I refer you to my articles on Fractals in Logo in the November and December 1985 issues of this magazine.)

A fractal curve is simple, yet infinitely complex — simple because it is made with a simple shape; infinitely complex because this simple shape is repeated many times, at smaller and smaller scales.

The classical mathematicians said that a straight line is considered to have one dimension, a flat plane has two dimensions and a volume of space has three. All very logical and neat. But, the maverick mathematicians challenged that notion and said between the first and second dimensions there was a continuous blending as one dimension blended into the next. This is the fractal area; between the first and second dimension.The term fractal was first coined by Benoit Mandelbrot as he attempted to explain the forms and chances that exist in nature. The term represents something that is irregular, broken, or fragmented, with recursive properties. I will be looking at only nonrandom mathematical fractals. When randomness is included in the fractal equation, the geometric beauty of the nonradom fractals changes to the organic beauty of natural form. The branching of trees, the meandering of streams, the awesome expanse of stars and galaxies throughout the heavens are all natural fractals.

The process of creating a fractal curve begins with a line segment (called an initiator). This line segment is replaced by a simple shape (called a generator), and each line segment in this generating shape is replaced by the same shape but on a smaller and smaller scale (called recursion). In theory, this replacement can continue infinitely, but in practice it can continue only as long as each line segment can be identified and until processing time becomes impractically long. Three to five stages (or levels) are usually good enough, but this will depend on the complexity of the curve itself.

Defining and measuring fractal curves is difficult. A fractal curve, by definition, has an infinite length, but it may surround a finite area. Mathematicians are not comfortable considering a fractal curve as a one dimensional entity, but it plainly is not two dimensional in nature. To solve this problem, the term fractional dimension was created to describe the particular niche between one and two dimensional entities that is occupied by fractal curves.

Fractal curves, therefore, are considered to have a dimensional value between one and two. Roughly, this dimension is a measure of the extent to which the fractal curve "fills" a two dimensional area. A fractal curve with a dimensional value of two completely fills the area (Hilbert and Sierpinski curves are two famous examples of such curves).

The program listings, both in the 8 bit as well as the 16 bit sections of this issue show four types of fractals. Each type has its own peculiarity and fascination.

PROGRAM LISTINGS

1. DRAGON SWEEP (listing for 800 series and ST.) Figures 3. This is a self contacting fractal, one that touches itself, but never crosses itself. It demonstrates in a step by step method how a fractal curve is filled. The generator consists of two line segments of equal length forming a right angle. During each relacement cycle, the generator is substituted for each segment on alternating sides of the segments.

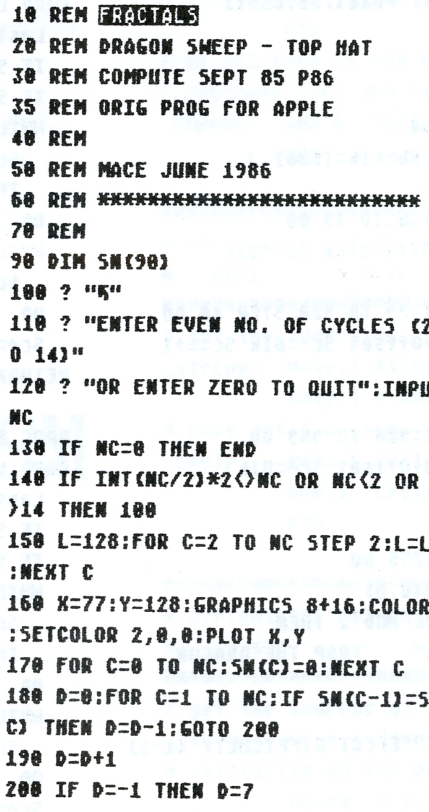
2. SIERPINSKI CURVE Versions 1, 2 and 3 (listings for 800 series. The ST version is in the library, version 1 only.) Figures 2. Version 1 is the straight curve. Run it on various levels, separately, or superimpose one level on another. Version 2 tilts the curve, apart from that the program runs the same way. Version 3 tilts and rounds (humps) the curve. Type in Version 1, then alter the necessary lines for versions 2 and 3.

3. SNOWFLAKE SWEEP (Listing for both 800 series and ST.) Figures 3. This is a self avoiding, self similar fractal. The generator was discovered by Mandelbrot and the resulting figure shows a line wandering around with tortouous complexity. The line is generated by the simple shape in fig 3.1 and it repeats itself with precise self similarity over a wide range of scales. Only the limitations of the screen resolution prevents the wiggles from getting smaller and smaller ad infinitum.

4. LAMBDA (listing for 800 series. The ST has this program in the public domain and is called Fractal1 by Leonard Tramiel.) See the library if you do not have it, or ask to have my listing included in another issue. This area was discovered (explored) by Greg Turk and his listing in Applesoft Basic appears in Byte, September 1984 page 170. Figures 4. A different variety of fractal results from looking at the behaviour of points in the complex plane (those points described by x+yi), where x and y are real numbers and i is the square root of minus one. (Refer to my article on Mandelbrots March 1986 for an explanation of the complex plane.) The equation that this program uses is f(z) = lambda * z * (1-z) where lambda and z are complex numbers. The function takes the point z and moves it to a new point f(z). The complex number lambda is a constant in this equation, and different values of lambda will result in different fractal curves. The program gives you a rough outline of the fractal curve by using the inverse equation. The program asks for the value of the constant lambda and, based on this and an initial value of z, it draws the position of z each time it is put through the function. Try these: lambda=3,0 (or 3.3,0), or 0,1, or 1,0. There are a whole universe of pictures you can generate depending on what you pick for lambda. When lambda equals two you get a circle.

I do have other fractal curves in the pipeline, all I need is time to get them working properly. I may do a follow up in a few months, by public demand only of course! Until then play around with these, change the number of segments the size of the initiator and other parts of the formula. Next month I hope to have an article put together on another fascinating area — chaotic (or stochastic) evolution. If anyone reading this has a mathematical background and has a few hours/days or months to spare to help me understand any of these areas I am exploring, you know my number! All help graciously accepted!

References: Mandelbrot - The fractal geometry of nature Fractals - P.R.Sorensen - Byte September 1984 p 157-172; Apple fractals - P.W.Carlson - Compute! September 1985 p 84-86; Fractals for the PC - Eui In Lee - PC Tech Journal p 165-178.



```
10 REM FRACTALS
20 REM DRAGON SWEEP - TOP HAT
30 REM COMPUTE SEPT 85 P86
35 REM ORIG PROG FOR APPLE
40 REM
50 REM MACE JUNE 1986
60 REM **************************
70 REM
90 DIM SN(90)
100 ? "↖"
110 ? "ENTER EVEN NO. OF CYCLES (2 T
0 14)"
120 ? "OR ENTER ZERO TO QUIT":INPUT
NC
130 IF NC=0 THEN END
140 IF INT(NC/2)*2<>NC OR NC<2 OR NC
>14 THEN 100
150 L=128:FOR C=2 TO NC STEP 2:L=L/2
:NEXT C
160 X=77:Y=128:GRAPHICS 8+16:COLOR 1
:SETCOLOR 2,0,0:PLOT X,Y
170 FOR C=0 TO NC:SN(C)=0:NEXT C
180 D=0:FOR C=1 TO NC:IF SN(C-1)=SN(
C) THEN D=D-1:GOTO 200
190 D=D+1
200 IF D=-1 THEN D=7
```

TRAP THE DRAGON BY STAN OCKERS

```
; ******************
; * TRAP THE DRAGON *
; *    S.O. 9-86    *
; ******************

DEFINE Blk="4",Ds="5",You="134"

INT ARRAY Nbor=[41 40 39 1 1 39 40 4
1]
BYTE Cset,CONSOL=53279,RTCLOK=19
CARD Scrad=88
BYTE POINTER Sc

PROC Init()
BYTE ARRAY Newchar=[162 170 138 170
 168 170 162 170 4 20 17 16 68 84 20
 0 255 255 255 255 255 255 255 255]
CARD Dlist=$230,J
  Cset=Peek(106)-4 Poke(106,Cset-1)
  MoveBlock(Cset*256,$E000,1024)
  MoveBlock(Cset*256+32,Newchar,24)
  Nbor(4)=-1 Nbor(5)=-39 Nbor(6)=-40
  Nbor(7)=-41
  Graphics(0) Poke(712,42) Poke(710,
54)
  Poke(709,34) Poke(708,212) Poke(71
1,134)
  FOR J=Dlist+6 TO dlist+28 DO
    Poke(J,4)
  OD
  Poke(752,1) Poke(756,Cset)
RETURN

PROC Bkgnd()
BYTE C,CH=764
CARD Offset,Nbrblk=[500],J
  Put(125)
  FOR Offset=0 TO 39 DO
  Sc=Scrad+Offset+40 Sc^=Blk
  OD
  FOR Offset=79 TO 920 STEP 40 DO
   Sc=Scrad+Offset Sc^=Blk Sc==+1
   Sc^=Blk
  OD
  FOR Offset=920 TO 959 DO
    Sc=Scrad+Offset Sc^=Blk
  OD
  DO
   WHILE CH=255 DO
    POSITION(8,0)
    IF RTCLOK MOD 2 THEN
     PRINT ("    TRAP THE DRAGON
")
    ELSE
     PRINT ("SELECT DIFFICULTY [1-5]
 ")
    FI
```

```
   OD
    C=GetD(7) C==-48
   UNTIL C>0 AND C<6
   OD
   POSITION(8,0)
   PRINT("   DIFFICULTY LEVEL = ")
   PRINTB(C)
   Nbrblk=(5-C)*75+100
   FOR J=1 TO Nbrblk DO
    Offset=Rand(0)*3+40 Offset==+Rand
 (194)
    Sc=Scrad+Offset Sc^=Blk
   OD
  RETURN

  PROC Delay(BYTE D)
  BYTE Cntclk=20,T
   DO T=Cntclk+D UNTIL T<256 OD
   DO UNTIL Cntclk=T OD
  RETURN

  CARD FUNC Move(CARD Scrloc, INT Offs
  et
       ,BYTE Shape)
   Sc=Scrloc Sc^=0 Sc==+Offset
   IF Sc^=0 THEN Sc^=Shape RETURN(Sc)

   ELSE Sc=Scrloc Sc^=Shape RETURN(Sc
  )
   FI

  PROC Bumpup(CARD Scrloc)
  CARD Lastsc
   Lastsc=Scrad+40 Sc=Scrloc-40
   IF Sc<Lastsc THEN RETURN FI
   IF Sc^<>Blk THEN RETURN FI
   WHILE Sc^<>0 DO
     Sc==-40
     IF Sc<Lastsc THEN RETURN FI
   OD
   WHILE Sc<Scrloc-40 DO
     Sc^=Blk Sc==+40
   OD
   Sc^=0
  RETURN

  PROC Bumpdn(CARD Scrloc)
  CARD Lastsc
   Lastsc=Scrad+920 Sc=Scrloc+40
   IF SC>Lastsc THEN RETURN FI
   IF Sc^<>Blk THEN RETURN FI
   WHILE Sc^<>0 DO
     Sc==+40
     IF Sc>Lastsc THEN RETURN FI
   OD
   WHILE Sc>Scrloc+40 DO
     Sc^=Blk Sc==-40
   OD
   Sc^=0
```

```
RETURN

PROC BumpIt(CARD Scrloc)
CARD Lastsc
  Sc=Scrloc-1
  Lastsc=Scrad+((Scrloc-Scrad)/40)*4
0
  IF Sc<Lastsc THEN RETURN FI
  IF Sc^<>Blk THEN RETURN FI
  WHILE Sc^<>0 DO
    Sc==-1
    IF Sc<Lastsc THEN RETURN FI
  OD
  WHILE Sc<Scrloc-1 DO
    Sc^=Blk Sc==+1
  OD
  Sc^=0
RETURN

PROC Bumprt(CARD Scrloc)
CARD Lastsc
  Sc=Scrloc+1
  Lastsc=Scrad+((Scrloc-Scrad)/40)*4
0+39
  IF Sc>Lastsc THEN RETURN FI
  IF Sc^<>Blk THEN RETURN FI
  WHILE Sc^<>0 DO
    Sc==+1
    IF Sc>Lastsc THEN RETURN FI
  OD
  WHILE Sc>Scrloc+1 DO
    Sc^=Blk Sc==-1
  OD
  Sc^=0
RETURN

BYTE FUNC Surr(CARD Scrloc,BYTE Test
)
BYTE J
  FOR J=0 TO 7 DO
    Sc=Scrloc+Nbor(J)
    IF Sc^=Test THEN RETURN(1) FI
  OD
  RETURN(0)

CARD FUNC MoveU(CARD Uloc)

  IF (Stick(0)&1)=0 THEN
    IF STrig(0) THEN
      Uloc=Move(Uloc,-40,You)
  ELSE Bumpup(Uloc)
   FI
  FI
  IF (Stick(0)&2)=0 THEN
    IF STrig(0) THEN
      Uloc=Move(Uloc,40,You)
    ELSE Bumpdn(Uloc)
    FI
  FI
```

```
 IF (Stick(0)&4)=0 THEN
   IF STrig(0) THEN
     Uloc=Move(Uloc,-1,You)
   ELSE BumpIt(Uloc)
   FI
 FI
 IF (Stick(0)&8)=0 THEN
   IF STrig(0) THEN
     Uloc=Move(Uloc,1,You)
   ELSE Bumprt(Uloc)
   FI
 FI
RETURN(Uloc)

CARD FUNC Movedrag(CARD Scrloc,BYTE
Shape)
BYTE J
INT Frst
  J=Rand(8) Sc=Scrloc Sc^=0
  Frst=Nbor(J) Sc==+Frst
  IF Sc^=0 THEN Sc^=Shape RETURN(Sc)
 FI
  DO
    J==+1 IF J>7 THEN J=0 FI
    Sc=Scrloc+Nbor(J)
    IF Sc^=0 THEN Sc^=Shape RETURN(S
c) FI
  UNTIL Nbor(J)=Frst
  OD
  Sc=Scrloc Sc^=Shape
RETURN(Sc)

PROC Lose()
BYTE T,J,V
  FOR J=150 TQ 230 STEP 4 DO
    FOR V=2 TO 10 STEP 2 DO
      Sound(0,J,10,12-V) Delay(1)
    OD
  OD
  SNDRST()
  DO Position(7,0)
   IF RTCLOK MOD 2 THEN
    Print("    Sorry... You Lose!
  ")
   ELSE
    Print(" Press START to play agai
n")
   FI
  UNTIL CONSOL=6
  OD
RETURN

PROC Win()
BYTE T,J,V
  FOR J=100 TO 200 STEP 10 DO
    FOR V=2 TO 10 STEP 2 DO
      SOUND(0,230-J,10,12-V) DELAY(1
)
    OD
  SNDRST()
  DO Position(7,0)
   IF RTCLOK MOD 2 THEN
    Print("Congratulations! You win!
  ")
   ELSE
    Print(" Press START to play agai
n")
   FI
  UNTIL CONSOL=6
  OD
RETURN

PROC Main()
CARD Drag,J,Upos
  Init()
  WHILE 1=1 DO
    Bkgnd()
    DO
     Drag=Scrad+Rand(255)+80
    UNTIL Peek(Drag)=0
    OD
    DO
      Upos=Scrad+400+Rand(255)
    UNTIL Peek(Upos)=0
    OD
    Poke(Upos,You)
    DO
      Drag=Movedrag(Drag,Ds)
      FOR J=1 TO 2000 DO OD
      Upos=MoveU(Upos)
    UNTIL
      (Surr(Upos,Ds)=1) OR (Surr(Dra
g,0)=0)
    OD
    IF Surr(Upos,Ds)=1 THEN Lose()
    ELSE Win()
    FI
    Poke(77,0)
  OD
RETURN
```

WALDEN'S ASM

16-BIT

```
* String functions for Alcyon

.globl  _strcat
.globl  _strcmp
.globl  _strcpy
.globl  _strlen

                .text

********************************
* strcat(str1,str2) - copy str2
*   to the end of str1
********************************
* get the length of the string
* will return with address in A0
_strcat:  bsr    _strlen
* backup to the 0 at end of string
          subq.l #1,a0
* get address of second string
          move.l 8(sp),a1
* copy the string
 catlp:   move.b (a1)+,(a0)+
          bne.s  catlp
 catend:  rts

********************************
* strcmp(str1,str2) compare strings
*   by subtracting str2 from
*   str1 and return difference
********************************
* get addresses of the strings
_strcmp:  move.l 4(sp),a0
          move.l 8(sp),a1
* get byte from each string
 strcmplp: move.b (a0)+,d0
* stop if end of string
          beq.s  cmpend
* subtract second string char
          sub.b  (a1)+,d0
* continue if the same
          beq.s  strcmplp
* else return the difference
          rts
* we get here is end of str1
* subtract str2 and return the diff
 cmpend:  sub.b  (a1),d0
          rts

********************************
* strcpy(str1,str2) copy str2 to
*   str1
********************************
* get the addresses
_strcpy:  move.l 4(sp),a0
          move.l 8(sp),a1
* move until a zero
 cpylp:   move.b (a1)+,(a0)+
          bne.s  cpylp:
          rts

********************************
* strlen(str) return the length
*   of str
********************************
* get the address of the string
_strlen:  move.l 4(sp),a0
* initialize D0 for DBcc loop
          moveq  #-1,d0       * ini
```

WALDEN CON'T

```
tialize length to -1 for loop
* test for zero
 lenlp:    tst.b   (a0)+
* adjust D0, stop if end of str
          dbeq     d0,lenlp
* make D0 positive
          neg      d0
* ignore zero at end of string
          subq     #1,d0
* return address+1 in A0
* return length in D0
          rts
********************************
```

RENEW NOW ==>

IF DATE EARLIER THAN NOV 86

THE AUSTRALIAN ATARI GAZETTE

```
10     rem EPICYCLOIDAL MICROCOMPUTER ART
15     rem        (520ST Version)
20     rem    by Andrev Plukss (MACE)
30     rem ------------------------------
40     fullv 2:clearv 2
50     dim Filename$(15),b$(4),file$(10)
60     print "Enter filename to save or load (8 characters maximum)"
70     print "(No quotes needed. Extender '.pic' added by program.)"
80     input file$
90     if len(file$)>8 then goto 60
100    let b$=".pic"
110    Filename$=file$+b$
120    defdbl Z:Z=SYSTAB+20
130    open "R",#1,Filename$
140    I%=LOF(1):close #1
150    if I%<>0 then goto PIC.READY
160    rem -----------------------
200    rem MICROART plotting routine
210    DEFSNG D,K-V
220    ?:? "Enter step size between 0.0001 and 0.01 (Smaller values";
230    ? "lead to finer detail but also decrease plotting speed.)"
240    INPUT D:rem D=stepsize
250    ?:? "Enter R,S,T,U,NA,NB,M,L,V,K"
260    INPUT R,S,T,U,NA,NB,M,L,V,K
270    fullv 2:clearv 2
280    PI=3.14159
290    N=NA/NB:Q=abs(NB+NB)
300    for A=0 to Q*PI+D step D
310    X=R*cos(A)+S*cos(N*A)+T*cos(M*A)+U*cos(L*A)+V*cos(K*A)
330    Y=R*sin(A)+S*sin(N*A)+T*sin(M*A)+U*sin(L*A)+V*sin(K*A)
340    X=X+X+307:Y=Y+Y+172
350    LINEF X,Y,X,Y
360    next A
500    rem -----------------------------
510    rem Screen save routine
520    reset:bsave Filename$,peek(Z),32767
530    end
1000   rem -----------------------------
1010   PIC.READY:
1020   rem Screen retrieval subroutine lines 1000-1050
1030   reset:bload Filename$,peek(Z)
1040   openv 2:
1050   if inp(2)=0 then 1050
1060   end
```

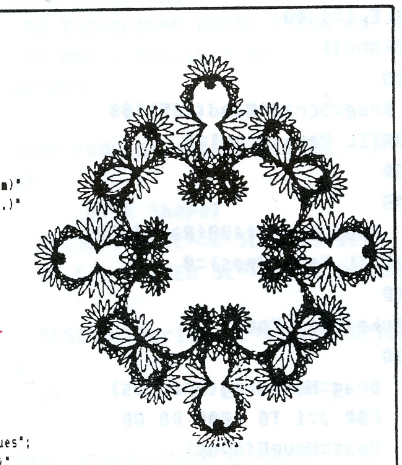


Figure 3.

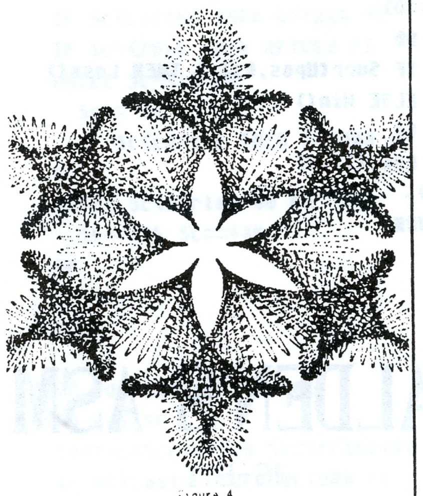


Figure 4.

ST

GETTING YOUR 8-BIT ATARI TO TALK TO YOUR ST

The following material was taken from the Mar.86 National Capital Atari User's Group and was written by Rick Beetham.

Many new owners of the Atari ST computers have probably come across the problem of transferring files from their old 8-Bit Atari computers. In most cases, getting the ST to talk to their modem was just a matter of purchasing or building a cable, acquiring a terminal package, and then calling up their local BBS. After the initial thrill of getting their ST up and running they probably started to notice that their is still a lot of good software available for their 8 Bit machine. All they have to do is download it... well maybe they will re-connect their 8 bit machine to the modem and call back. Of course that is always the time when they find that elusive ST subroutine that they have been looking for so then they have to disconnect, reconnect their modem to the ST, recall the BBS, then....

And of course, what about the disk subscription you have to Antic magazine? How the heck are you going to get those ST programs off of that 5 1/4" disk? Do you really have to upload all those files to the local BBS with your 8 bit then call back with your ST and download them back onto your 16 bit machine? Sigh.... if only you had bought that 850 interface.... etc.,etc.

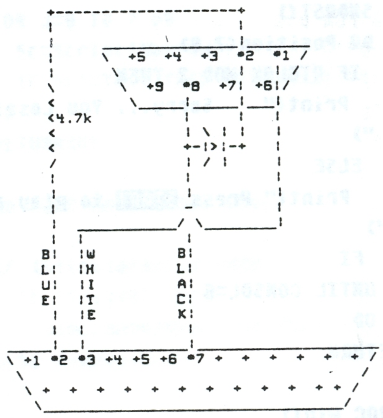
Well, here is an easier way to transfer files and you don't need an 850 interface; or two modems; or a friendly SYSOP in order to do it. The solution is using a null modem cable. The cable gives you a very bare-bones RS232 configuration of a signal ground, a receive data, and a transmit data line. In order to make a null-modem all one has to do is connect the receive data line from the 8 bit to the transmit data line of the 16 bit; likewise the transmit data line from the 8 bit is connected to the receive data line on the 16 bit; signal ground is common to both.

After building the null-modem I then used STTERM2 on my ST and Smartterm 5.0 on my 8 bit. I set both terminals to Xmodem protocol, 1200 baud, ASCII, and half duplex. I successfully transfered several test files from one Atari to the other without fail.

The cost for this cable will be all of about $13, and that is if you buy all your parts new from Radio Shack. If you have an old Atari joystick, you can use the cable from that and save even more.

PARTS LIST:

Atari joystick, plug & cord	- 2761538
Length of 3 strand wire	
Switching diode	- 2761122
4.7k 1/2 watt resistor	- 2718030
25 pin RS232 female connector	- 2761548



Note: Black and White wires do not Connect

FRACTALS CON'T

```
210 IF D=8 THEN D=0
220 NEXT C
230 IF D=0 THEN X=X+L:GOTO 270
240 IF D=2 THEN Y=Y+L:GOTO 270
250 IF D=4 THEN X=X-L:GOTO 270
260 Y=Y-L
270 DRAWTO X,Y:SN(NC)=SN(NC)+1
280 FOR C=NC TO 1 STEP -1:IF SN(C)<>
2 THEN 300
290 SN(C)=0:SN(C-1)=SN(C-1)+1:NEXT C

300 IF SN(0)=0 THEN 180
320 GOTO 320
```

```
1 REM SNOWFLAKE SWEEP - FRACTAL
2 REM
3 REM MACE JUNE 1986
4 REM ************************
5 REM
10 CLR :? "K":SETCOLOR 2,0,0:REM PRO
GRAM 3
20 REM ORIGINAL PROGRAM FOR APPLE
30 REM THIS PROGRAM PLOTS A FRACTAL
'SNOWFLAKE SWEEP'
40 ? "WHEN DRAWING IS FINISHED":? :?
 "PRESS START FOR ANOTHER DRAWING.":
? "PRESS SELECT TO SAVE PICTURE"
50 DIM A$(1),FILE$(15),DX(11),DY(11)
,SD(6),RD(6),LN(6),SN(4):M=7/6
55 POSITION 2,7:? "ENTER DEVICE FILE
NAME TO SAVE PICTURE (ENTER C. FOR C
ASSETTE)":INPUT FILE$:IF FILE$="" TH
EN 55
60 FOR N=0 TO 6:READ S,R:SD(N)=S:RD(
N)=R:LN(N)=1/3:NEXT N:LN(2)=SQR(LN(1
))
70 A=0:FOR D=6 TO 11:DX(D)=COS(A):DY
(D)=SIN(A)
80 A=A+0.52359879:NEXT D
90 FOR D=0 TO 5:DX(D)=-DX(D+6):DY(D)
=-DY(D+6):NEXT D
100 GRAPHICS 0
110 ? "KENTER NUMBER OF CYCLES (1 -
4)        OR ENTER ZERO TO QUIT"
120 INPUT NC
130 IF NC=0 THEN END
140 IF NC>4 THEN 110
150 GRAPHICS 24:COLOR 1:SETCOLOR 2,0
,0
155 PLOT 0,0:DRAWTO 319,0:DRAWTO 319
,191:DRAWTO 0,191:DRAWTO 0,0
160 X=255:Y=142:TL=162:PLOT X,Y
170 FOR C=0 TO NC:SN(C)=0:NEXT C:D=0
:L=TL:NS=0
180 FOR C=1 TO NC:I=SN(C):L=L*LN(I):
J=SN(C-1):NS=NS+SD(J):K=INT(NS/2):IF
 K*2<>NS THEN D=D+12-RD(I):GOTO 200
190 D=D+RD(I)
200 IF D>11 THEN D=D-12
```

```
210 NEXT C
220 X=X+M*L*DX(D):Y=Y-L*DY(D):DRAWTO
 X,Y:SN(NC)=SN(NC)+1:FOR C=NC TO 1 S
TEP -1:IF SN(C)<>7 THEN 240
230 SN(C)=0:SN(C-1)=SN(C-1)+1:NEXT C

240 POKE 77,0:IF SN(0)=0 THEN D=0:L=
TL:NS=0:GOTO 180
250 IF PEEK(53279)=6 THEN 10
255 IF PEEK(53279)=5 THEN 300
260 GOTO 250
270 DATA 0,0,1,0,1,7,0,10,0,0,0,2,1,
2
300 CLOSE #1:OPEN #1,8,0,FILE$
310 S=PEEK(88)+256*PEEK(89):TOP=S+76
80:BS=TOP-S
320 HI=INT(BS/256):LO=BS-(HI*256):PO
KE 850,11:POKE 852,PEEK(88):POKE 853
,PEEK(89):POKE 856,LO:POKE 857,HI
330 D=USR(ADR("hhh□LV□"),16):CLOSE #
1:GOTO 10
```

```
1 REM GREG TURK'S FRACTAL INVESTIGAT
IONS
2 REM FRACTALS IN THE COMPLEX PLANE
3 REM
4 REM MACE JUNE 1986
5 REM ******************************
*
6 REM
10 ? "K":? "                  FRACTALS"

15 ? " REFER BYTE MAGAZINE - SEPTEMB
ER 1984":?
20 ? "PRESS OPTION FOR SCREEN DUMP":
? :? "PRESS SELECT TO ENTER NEW PARA
METERS  WITHOUT LOSING PICTURE":?
30 ? "SUGGEST X,Y VALUES BETWEEN 0 A
ND 5 ANDSCALE BETWEEN 1 AND 10.  TRY
 SCALE OF 5 TO START."
40 ? "THE SMALLER THE SCALE NUMBER T
HE      LARGER THE PICTURE":? "TRY X
=3. Y=0. SCALE=2":? "    X=1. Y=0. S
CALE=5"
50 ? "    X=0. Y=1. SCALE=5":? "PRES
S START TO START BEGIN NEW DRAWING"
60 IF PEEK(53279)=6 THEN 80
70 GOTO 60
80 IF PEEK(53279)<>7 THEN 80
90 DIM A$(192)
100 GRAPHICS 8:COLOR 1:SETCOLOR 2,0,
0:CX=160:CY=75
110 X=0.50001:Y=0
120 GOSUB 430
130 FOR I=1 TO 10:GOSUB 350:NEXT I
140 GOSUB 510
150 GOSUB 350
160 GOTO 140
170 END
```

```
180 REM SQUARE ROOT OF X,Y
190 T=Y
200 S=SQR(X*X+Y*Y)
210 Y=SQR(ABS((-X+S)/2))
220 X=SQR(ABS((X+S)/2))
230 IF T<0 THEN X=-X
240 RETURN
250 REM FOUR OVER L
260 S=LX*LX+LY*LY
270 LX=4*LX/S
280 LY=-4*LY/S
290 RETURN
300 REM X,Y TIMES L
310 TX=X:TY=Y
320 X=TX*LX-TY*LY
330 Y=TX*LY+TY*LX
340 RETURN
350 REM FUNCTION OF X,Y
360 GOSUB 300
370 X=1-X
380 GOSUB 180
390 IF RND(0)<0.5 THEN X=-X:Y=-Y
400 X=1-X
410 X=X/2:Y=Y/2
420 RETURN
430 REM GET VALUES
440 ? "WHAT IS LAMBDA?(X,Y)";
450 INPUT LXI,LYI:LX=LXI:LY=LYI
460 GOSUB 250
470 ? "WHAT IS SCALE";:INPUT SCI:SC=
SCI:? :? :? :?
480 POKE 656,0:? "LAMBDA(X/Y): X=";L
XI;": ";"Y=";LYI;"   ":? "SCALE=";SC
I;"    "
490 SC=2*CX/SC
500 RETURN
510 REM PLOT X,Y
520 XF=INT(SC*(X-0.5)+CX):YF=INT(CY-
SC*Y)
530 POKE 752,1:POKE 656,2:? "XF=";XF
;"  ","YF=";YF;" "
540 IF XF<0 THEN XF=0
550 IF XF>319 THEN XF=319
560 IF YF<0 THEN YF=0
570 IF YF>159 THEN YF=159
580 PLOT XF,YF
590 IF PEEK(53279)=3 THEN GOSUB 630
600 IF PEEK(53279)=5 THEN ? :? :? :P
OKE 656,0:GOTO 430
610 IF PEEK(53279)=6 THEN 100
620 POKE 77,0:RETURN
630 RESTORE 690:LPRINT CHR$(27);CHR$
(64):LPRINT "LAMBDA(X/Y). X=";LXI;"
 Y=";LYI:LPRINT "SCALE=";SCI
640 FOR B=1 TO 61:READ N:POKE 1535+B
,N:NEXT B:DM=PEEK(88)+PEEK(89)*256:D
M=DM+40*159
650 LPRINT CHR$(27);"3";CHR$(20):FOR
 V=DM TO DM+39
```

```
660 A$=CHR$(0):A$(160)=CHR$(0):A$(2)
=A$
670 W=USR(1536,V,ADR(A$)):LPRINT CHR
$(27);"K";CHR$(160);CHR$(0);A$
680 NEXT V
690 DATA 104,104,141,21,6,104,141,20
,6,104,141,27,6,104,141,26,6,160,193
,173,255,255,136,240,35,141,255,255,
238
700 DATA 26,6,240,21,173,20,6,56,233
,40,141,20,6,144,4,24,76,19,6,206,21
,6,76,19,6,238,27,6,76,33,6,96
710 RETURN
```

```
1 REM SIERPINSKI CURVES
2 REM CREATIVE COMPUTING JULY 84(P15
0)
3 REM SCIENTIC AMERICAN SEPT 1976
4 REM
5 REM MACE JUNE 1986
6 REM ******************************
*
7 REM
8 DIM ST(100)
9 GOSUB 900
10 GRAPHICS 8+16:SETCOLOR 2,0,0
15 COLOR 1
25 REM
30 REM SET LEVEL OF CURVE
```

```
35 REM
40 FOR DI=01 TO 02
60 GOSUB 100
80 NEXT DI
90 GOTO 90
94 REM
95 REM INITIALIZATION AND MAIN CURVE

96 REM
100 H0=192:SP=0:H=H0/4:X=2*H:Y=3*H:I
=0
110 I=I+1:X=X-H:H=H/2:Y=Y+H
120 IF I<DI THEN 110
130 PS=I:GOSUB 600
140 GOSUB 200:A=H:B=-H:GOSUB 800
150 GOSUB 300:A=-H:B=-H:GOSUB 800
160 GOSUB 400:A=-H:B=H:GOSUB 800
170 GOSUB 500:A=H:B=H:GOSUB 800
180 GOSUB 700
190 RETURN
195 REM
200 REM SUBROUTINE A
210 IF TP<=0 THEN RETURN
220 PS=TP-1:GOSUB 600
230 GOSUB 200:A=H:B=-H:GOSUB 800
240 GOSUB 300:A=2*H:B=0:GOSUB 800
250 GOSUB 500:A=H:B=H:GOSUB 800
260 GOSUB 200
270 GOSUB 700
280 RETURN
295 REM
300 REM SUBROUTINE B
310 IF TP<=0 THEN RETURN
320 PS=TP-1:GOSUB 600
330 GOSUB 300:A=-H:B=-H:GOSUB 800
340 GOSUB 400:A=0:B=-2*H:GOSUB 800
350 GOSUB 200:A=H:B=-H:GOSUB 800
360 GOSUB 300
370 GOSUB 700
380 RETURN
395 REM
400 REM SUBROUTINE C
410 IF TP<=0 THEN RETURN
420 PS=TP-1:GOSUB 600
430 GOSUB 400:A=-H:B=H:GOSUB 800
440 GOSUB 500:A=-2*H:B=0:GOSUB 800
450 GOSUB 300:A=-H:B=-H:GOSUB 800
460 GOSUB 400
470 GOSUB 700
480 RETURN
495 REM
500 REM SUBROUTINE D
510 IF TP<=0 THEN RETURN
520 PS=TP-1:GOSUB 600
530 GOSUB 500:A=H:B=H:GOSUB 800
540 GOSUB 200:A=0:B=2*H:GOSUB 800
550 GOSUB 400:A=-H:B=H:GOSUB 800
560 GOSUB 500
570 GOSUB 700
```

THE AUSTRALIAN ATARI GAZETTE

```
10 REM EPICYCLOIDAL MICROCOMPUTER ART
11 REM (800/800XL/130XE Version)
12 REM by Andrew Plukss (MACE)
13 REM ------------------------------
15 GRAPHICS 0:CLR :DIM FILE$(15)
20 ? :? "} When pattern has finished, pressing   OPTIO
N will save screen display to    disk."
25 ? :? "Enter Filename just in case you wish  to save
 the screen later as"
30 ? :? "   D:FILENAME.EXT (no quotes)":INPUT FILE$
160 REM ----------------------------
200 REM Plotting routine
220 ? "}Enter step size between .0001 and .01":? "Comp
lex patterns require a small step size."
240 INPUT D:REM D is step size
250 ? :? "Enter R,S,T,U,NA,NB,M,L,V,K"
260 INPUT R,S,T,U,NA,NB,M,L,V,K
270 GRAPHICS 24:COLOR 1:SETCOLOR 2,0,0
280 PI=3.14159
290 N=NA/NB:Q=ABS(NB+NB)
300 FOR A=0 TO Q*PI+D STEP D
310 X=R*COS(A)+S*COS(N*A)+T*COS(M*A)+U*COS(L*A)+V*COS(
K*A)
330 Y=R*SIN(A)+S*SIN(N*A)+T*SIN(M*A)+U*SIN(L*A)+V*SIN(
K*A)
340 X=X+160:Y=Y+96
345 IF A=0 THEN PLOT X,Y
350 DRAWTO X,Y
351 POKE 77,0:REM Turns off attract mode
360 NEXT A
370 TRAP 40000:IF PEEK(53279)=3 THEN 520:REM Checks if
 OPTION pressed
380 GOTO 370
500 REM ----------------------------
510 REM Screen save routine
520 CLOSE #1:OPEN #1,8,0,FILE$
530 S=PEEK(88)+256*PEEK(89):TOP=S+7680:BS=TOP-S
540 HI=INT(BS/256):LO=BS-(HI*256):POKE 850,11:POKE 852
,PEEK(88):POKE 853,PEEK(89):POKE 856,LO:POKE 857,HI
550 D=USR(ADR("hhh*LVd"),16):REM * and d should be typ
ed in inverse
560 CLOSE #1:GOTO 10
```

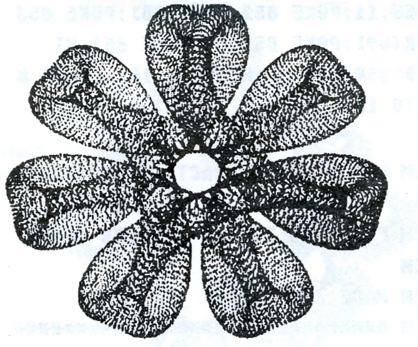


Figure 1.

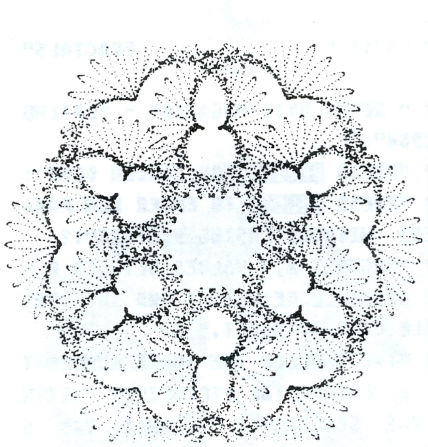


Figure 2.

REFER TO STING SECTION
FOR ST PROGRAM ON PAGE 79

A few patterns to try out:

R	S	T	U	NA	NB	M	L	V	K	step size	
45	27	6	-13.8	-5	2	-207.5	202.5	0	0	.0005	(figure 1)
56	28	7	6.3	7	1	331	-299	0	0	.0005	(figure 2)
47.5	15	15	4	-7	1	13	622	3.5	-588	.0005	(figure 3)
60	24	14.4	13.6	-5	1	379	-365	0	0	.0005	(figure 4)
45	30	12	0	-54	1	110	0	0	0	.005	
40	40	0	0	7	1	0	0	0	0	.1	
30	30	0	0	-12	5	0	0	0	0	.025	
40	20	15	0	-8	1	-26	0	0	0	.01	
30	30	-8	0	-3	5	-10.2	0	0	0	.025	

FRACTALS CON'T

```
580 RETURN
595 REM
600 REM PUSH SUBROUTINE
610 SP=SP+1:ST(SP)=PS
620 TP=PS:RETURN
695 REM
700 REM POP SUBROUTINE
710 SP=SP-1:TP=ST(SP):RETURN
795 REM
800 REM PLOT SUBROUTINE
810 PLOT Y+65,X:DRAWTO Y+65+B,X+A
820 X=X+A:Y=Y+B:RETURN
900 REM INSTRUCTION SUBROUTINE
910 ? CHR$(125)
915 POSITION 11,0
920 ? "SIERPINSKI CURVES"
925 ? :? "     ADAPTED FOR THE ATARI
 FROM ":? "   CREATIVE   COMPUTING J
ULY 1984"
930 ? :? "FOR A FIRST ORDER CURVE  1
,1"
940 ? :? "FOR A SECOND ORDER CURVE 2
,2"
950 ? :? "FOR A THIRD ORDER CURVE  3
,3"
955 ? :? "FOR A FOURTH ORDER CURVE 4
,4"
960 ? :? "FOR A FIFTH ORDER CURVE  5
,5"
965 ? :? " FOR SUPERMIPOSED CURVES E
NTER THE      LOWEST ORDER FIRST"
990 TRAP 990:POSITION 16,21:? "
"
1000 POSITION 2,20:? "ENTER THE FIRS
T AND LAST ORDER CURVE  TO BE PLOTTE
D";:INPUT O1,O2
1005 IF O1<1 OR O1>5 OR O2<1 OR O2>5
 OR O1>O2 THEN 990
1010 TRAP 40000:RETURN
```

```
 1 REM SIERPINSKI CURVE - TILTED
 2 REM CREATIVE COMPUTING JULY 84(P15
0)
 3 REM SCIENTIC AMERICAN SEPT 1976
 4 REM
 5 REM MACE JUNE 1986
 6 REM ******************************
*
 7 REM
 8 GOSUB 900
 9 DIM ST(100)
 10 GRAPHICS 8+16:SETCOLOR 2,0,0
 15 COLOR 1
 25 REM
 30 REM SET LEVEL OF CURVE
 35 REM
 40 FOR DI=O1 TO O2
 60 GOSUB 100
```

```
80 NEXT DI
90 GOTO 90
94 REM
95 REM INITIALIZATION AND MAIN CURVE

96 REM
100 H0=319:SP=0:H=H0/4:X=2*H:Y=3*H:I
=0
110 I=I+1:X=X-H:H=H/2:Y=Y+H
115 YP=SQR(Y)*7
116 XP=X*(-YP/(1*H0)+1)+YP/2
120 IF I<DI THEN 110
130 PS=I:GOSUB 600
140 GOSUB 200:A=H:B=-H:GOSUB 800
150 GOSUB 300:A=-H:B=-H:GOSUB 800
160 GOSUB 400:A=-H:B=H:GOSUB 800
170 GOSUB 500:A=H:B=H:GOSUB 800
180 GOSUB 700
190 RETURN
195 REM
200 REM SUBROUTINE A
210 IF TP<=0 THEN RETURN
220 PS=TP-1:GOSUB 600
230 GOSUB 200:A=H:B=-H:GOSUB 800
240 GOSUB 300:A=2*H:B=0:GOSUB 800
250 GOSUB 500:A=H:B=H:GOSUB 800
260 GOSUB 200
270 GOSUB 700
280 RETURN
295 REM
300 REM SUBROUTINE B
310 IF TP<=0 THEN RETURN
320 PS=TP-1:GOSUB 600
330 GOSUB 300:A=-H:B=-H:GOSUB 800
340 GOSUB 400:A=0:B=-2*H:GOSUB 800
350 GOSUB 200:A=H:B=-H:GOSUB 800
360 GOSUB 300
370 GOSUB 700
380 RETURN
395 REM
400 REM SUBROUTINE C
410 IF TP<=0 THEN RETURN
420 PS=TP-1:GOSUB 600
430 GOSUB 400:A=-H:B=H:GOSUB 800
440 GOSUB 500:A=-2*H:B=0:GOSUB 800
450 GOSUB 300:A=-H:B=-H:GOSUB 800
460 GOSUB 400
470 GOSUB 700
480 RETURN
495 REM
500 REM SUBROUTINE D
510 IF TP<=0 THEN RETURN
520 PS=TP-1:GOSUB 600
530 GOSUB 500:A=H:B=H:GOSUB 800
540 GOSUB 200:A=0:B=2*H:GOSUB 800
550 GOSUB 400:A=-H:B=H:GOSUB 800
560 GOSUB 500
570 GOSUB 700
```

```
580 RETURN
595 REM
600 REM PUSH SUBROUTINE
610 SP=SP+1:ST(SP)=PS
620 TP=PS:RETURN
695 REM
700 REM POP SUBROUTINE
710 SP=SP-1:TP=ST(SP):RETURN
795 REM
800 REM PLOT SUBROUTINE
805 X=X+A:Y=Y+B
810 YQ=SQR(Y)*7
812 XQ=X*(-YP/(1*H0)+1)+YP/2
815 PLOT 319-XP,191-(YP*1.5):DRAWTO
319-XQ,191-(YQ*1.5)
820 XP=XQ:YP=YQ:RETURN
900 REM INSTRUCTION SUBROUTINE
910 ? CHR$(125)
915 POSITION 11,0
920 ? "SIERPINSKI CURVES"
925 ? :? "          ADAPTED FOR THE ATA
RI":? " FROM CREATIVE COMPUTING JULY
 1984"
930 ? :? "FOR A FIRST ORDER CURVE EN
TER  1,1"
940 ? :? "FOR A SECOND ORDER CURVE E
NTER 2,2"
950 ? :? "FOR A THIRD ORDER CURVE EN
TER  3,3"
955 ? :? "FOR A FOURTH ORDER CURVE E
NTER 4,4"
960 ? :? "FOR A FIFTH ORDER CURVE EN
TER  5,5"
965 ? :? "FOR OVERLAYED CURVES ENTER
 LOWEST     ORDER FIRST ":? :? "ORDE
R MUST BE BETWEEN 1 AND 5 INC"
999 TRAP 999:POSITION 7,21:? "      "
;
1000 POSITION 2,19:? "ENTER THEN FIR
ST AND LAST ORDER CURVE TO BE PLOTTE
D. ENTER THE LOWEST ORDER FIRST";
1001 INPUT O1,O2
1002 IF O1<1 OR O1>5 OR O2<1 OR O2>5
 OR O1>O2 THEN 999
1010 TRAP 40000:RETURN
```

```
1 REM SIERPINSKI CURVE - ROUNDED
2 REM CREATIVE COMPUTING JULY 84(P15
0)
3 REM SCIENTIC AMERICAN SEPT 1976
4 REM
5 REM MACE JUNE 1986
6 REM ******************************
*
7 REM
8 GOSUB 900
9 DIM ST(100)
10 GRAPHICS 8+16:SETCOLOR 2,0,0
15 COLOR 1
```

FRACTALS CON'T

```
25 REM
30 REM SET LEVEL OF CURVE
35 REM
40 FOR DI=01 TO 02
60 GOSUB 100
80 NEXT DI
90 GOTO 90
94 REM
95 REM INITIALIZATION AND MAIN CURVE

96 REM
100 H0=319:SP=0:H=H0/4:X=2*H:Y=3*H:I
=0
110 I=I+1:X=X-H:H=H/2:Y=Y+H
115 YP=SQR(Y)*7
116 XP=X*(-Y/(2*H0)+1)+Y/4
120 IF I<DI THEN 110
130 PS=I:GOSUB 600
140 GOSUB 200:A=H:B=-H:GOSUB 800
150 GOSUB 300:A=-H:B=-H:GOSUB 800
160 GOSUB 400:A=-H:B=H:GOSUB 800
170 GOSUB 500:A=H:B=H:GOSUB 800
180 GOSUB 700
190 RETURN
195 REM
200 REM SUBROUTINE A
210 IF TP<=0 THEN RETURN
220 PS=TP-1:GOSUB 600
230 GOSUB 200:A=H:B=-H:GOSUB 800
240 GOSUB 300:A=2*H:B=0:GOSUB 800
250 GOSUB 500:A=H:B=H:GOSUB 800
260 GOSUB 200
270 GOSUB 700
280 RETURN
295 REM
300 REM SUBROUTINE B
310 IF TP<=0 THEN RETURN
320 PS=TP-1:GOSUB 600
330 GOSUB 300:A=-H:B=-H:GOSUB 800
340 GOSUB 400:A=0:B=-2*H:GOSUB 800
350 GOSUB 200:A=H:B=-H:GOSUB 800
360 GOSUB 300
370 GOSUB 700
380 RETURN
395 REM
400 REM SUBROUTINE C
410 IF TP<=0 THEN RETURN
420 PS=TP-1:GOSUB 600
430 GOSUB 400:A=-H:B=H:GOSUB 800
440 GOSUB 500:A=-2*H:B=0:GOSUB 800
450 GOSUB 300:A=-H:B=-H:GOSUB 800
460 GOSUB 400
470 GOSUB 700
480 RETURN
495 REM
500 REM SUBROUTINE D
510 IF TP<=0 THEN RETURN
520 PS=TP-1:GOSUB 600
530 GOSUB 500:A=H:B=H:GOSUB 800
```

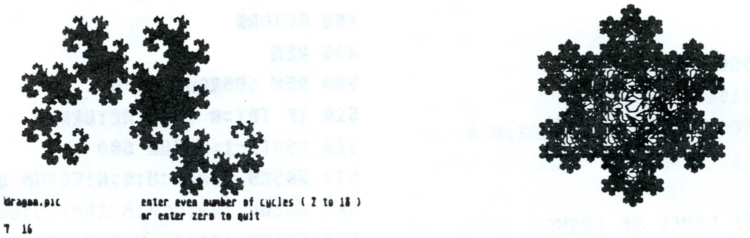
ST PROGRAMS

```
List of \DRAGON1.BAS
List of \DRAGON1.BAS

   5      '  MACE June 1986
   6      ' ****************
   10     ' dragon sweep - Fractal geometry of nature -
   20     ' Mandelbrot page 68   (Peano curve)
   30     ' also see  compute sept 85 p86 (for apple)
   40     ' rita
   50     ' ---------------------------------------
   60     ' set up screensave and retrieval routines
   70     ' ---------------------------------------
   80     fullw 2: clearw 2
   90     dim filename$(15),b$(4),file$(10)
   100    ?"enter name of screen to SAVE or LOAD - max 8 char."
   110    input "no quotes required.  PIC extender added by program ";fil
e$
   120    if len(file$)>8 then 100
   130    let b$=".pic"
   140    filename$=file$+b$
   150    defdbl P:P=SYSTAB+20
   160    open "R",#1,filename$
   170    i%=lof(1)
   180    close 1
   190    if i%<>0 then goto pic.ready
   200    '------------------------
   210    ' PROGRAM
   220    '------------------------
   230    dim sn(90)
   240    fullw 2: clearw 2
   250    ?filename$ ;
   260    ?"              enter even number of cycles ( 2 to 18 )
   270    ?"              or enter zero to quit ":input nc
   280    if nc=0 then end
   290    if int(nc/2)*2<>nc or nc<2 or nc>18 then 240
   300    ' m determines the length of the line
   310    m=228:for c= 2 to nc step 2:m=m/2:next c
   320    for c = 0 to nc:sn(c)=0:next c
   330    'd gives generator shape
   340    d=0:for c= 1 to nc:if sn(c-1)=sn(c) then d=d-1:goto 360
   350    d=d+1
   360    if d=-1 then d=7
   370    if d=8 then d=0
   380    next c
   390    if d=0 then x1=x+m:goto 430
   400    if d=2 then y1=y+m:goto 430
   410    if d=4 then x1=x-m:goto 430
   420    y1=y-m
   430    linef x+100 ,y+200    ,x1+100,y1+200 :sn(nc)=sn(nc)+1
   440    x=x1:y=y1
   450    for c=nc to 1 step -1:if sn(c)<>2 then 470
   460    sn(c)=0:sn(c-1)=sn(c-1)+1:next c
   470    if sn(0)=0 then 340

              480    '----------------------
              490    ' screensave routine
              500    '----------------------
              510    reset:bsave filename$,peek (P),32767
              520    END
              530    '----------------------
              540    ' screenload routine
              550    '----------------------
              560    pic.ready:
              570    reset:bload filename$,peek(P)
              580    openw 2:
              590    if inp(2)=0 then 590
              600    end
```

OUTPUT



```
dragon.pic                 enter even number of cycles ( 2 to 18 )
                           or enter zero to quit
7  16
```

FRACTELS CON'T

```
540 GOSUB 200:A=0:B=2*H:GOSUB 800
550 GOSUB 400:A=-H:B=H:GOSUB 800
560 GOSUB 500
570 GOSUB 700
580 RETURN
595 REM
600 REM PUSH SUBROUTINE
610 SP=SP+1:ST(SP)=PS
620 TP=PS:RETURN
695 REM
700 REM POP SUBROUTINE
710 SP=SP-1:TP=ST(SP):RETURN
795 REM
800 REM PLOT SUBROUTINE
805 X=X+A:Y=Y+B
810 YQ=SQR(Y)*7
812 XQ=X*(-Y/(2*H0)+1)+Y/4
815 PLOT 319-XP,191-(YP*1.5):DRAWTO
319-XQ,191-(YQ*1.5)
820 XP=XQ:YP=YQ:RETURN
900 REM INSTRUCTION SUBROUTINE
910 ? CHR$(125)
915 POSITION 11,0
920 ? "SIERPINSKI CURVES"
925 ? :? "     ADAPTED FOR THE ATARI
 FROM":? "   CREATIVE   COMPUTING JU
LY 1984"
930 ? :? "FOR A FIRST ORDER CURVE EN
TER  1,1"
940 ? :? "FOR A SECOND ORDER CURVE E
NTER 2,2"
950 ? :? "FOR A THIRD ORDER CURVE EN
TER  3,3"
955 ? :? "FOR A FOURTH ORDER CURVE E
NTER 4,4"
960 ? :? "FOR A FIFTH ORDER CURVE EN
TER  5,5"
965 ? :? "FOR OVERLAYED CURVES ENTER
 LOWEST     ORDER FIRST ":? :? "ORDE
R MUST BE BETWEEN 1 AND 5 INC"
999 TRAP 999:POSITION 7,21:? "      "
;
1000 POSITION 2,19:? "ENTER THEN FIR
ST AND LAST ORDER CURVE TO BE PLOTTE
D. ENTER THE LOWEST ORDER FIRST";
1001 INPUT O1,O2
1002 IF O1<1 OR O1>5 OR O2<1 OR O2>5
 OR O1>O2 THEN 999
1010 TRAP 40000:RETURN
```

RENEW
NOW
==>

IF DATE EARLIER

THAN NOV 86

```
List of \SNOWFRAC.BAS

   5      ' MACE  June 1986
   6      ' ****************
   10     ' SNOWFLAKE SWEEP
   20     ' Mandelbrot - Fractal geometry of nature
   30     ' also  compute! sept 85 (for apple)
   40     ' rita
   50     ' -----------------------------------------
   60     ' setup  screen save & retrieval routines
   70     ' -----------------------------------------
   80     fullw 2:clearw 2:color 1,1,1,1,1
   90     dim filename$(15),b$(4),file$(10)
   100    ?"enter name of screen to SAVE or LOAD - max of 8 characters"
   110    input "no quotes needed. PIC extender added by program ";file$
   120    if len(file$)>8 then 100
   130    let b$=".pic"
   140    filename$=file$+b$
   150    defdbl P:P=SYSTAB+20
   160    open "R",#1,filename$
   170    i%=lof(1)
   180    close 1
   190    if i%<>0 then goto pic.ready

   200    '------------------------
   210    ' PROGRAM
   220    '------------------------
   221    DIM DX(11),DY(11),SD(6),RD(6),LN(6),SN(4)
   225    M=7/6
   230    FOR N=0 to 6:READ S,R: SD(N)=S:RD(N)=R:LN(N)=1/3:NEXT N:LN(2)=S
QR(LN(1))
   235    A=0
   240    FOR D=6 TO 11:DX(D)=COS(A):DY(D)=SIN(A)
   250    A=A+0.52359879 :NEXT D
   260    FOR D=0 TO 5:DX(D)=-DX(D+6):DY(D)=-DY(D+6):NEXT D
   270    FULLW 2: CLEARW 2: COLOR 1,1,1,1,1
   280    ?"ENTER NUMBER OF CYCLES 1-4 OR ZERO TO QUIT ";:INPUT NC
   290    IF NC=0 THEN END
   300    IF NC>4 THEN 280
   310    X=400:Y=200:TL=162:LINEF X,Y,X,Y
   320    FOR C=0 TO NC:SN(C)=0:NEXT C:D=0:L=TL:NS=0
   330    FOR C=1 TO NC:I=SN(C):L=L*LN(I):J=SN(C-1):NS=NS+SD(J)
   340    K=INT(NS/2):IF K*2<>NS THEN D=D+12-RD(I):GOTO 360
   350    D=D+RD(I)
   360    IF D>11 THEN D=D-12
   370    NEXT C
   380    X2=X+M*L*DX(D):Y2=Y-L*DY(D):LINEF X,Y,X2,Y2
   385    X=X2:Y=Y2
   390    SN(NC)=SN(NC)+1
   400    FOR C=NC TO 1 STEP -1: IF SN(C)<>7 THEN 420
   410    SN(C)=0: SN(C-1)=SN(C-1)+1: NEXT C
   420    IF SN(0)=0 THEN D=0: L=TL:NS=0: GOTO 330
   430    DATA 0,0,1,0,1,7,0,10,0,0,0,2,1,2
   1000   ' ---- screen save routine -----
   1010   reset:bsave filename$,peek(P),32767
   1020   END
   1030   ' ---- screen load routine -----
   1040   pic.ready:
   1050   reset :bload filename$,peek(P)
   1060   openw 2:
   1070   if inp(2)=0 then 1070
   1080   end
```

BBS NEWS

This has been a rough month for the BBS. 5 hours after I left for a 10 day vacation, the hard disk crashed leaving the BBS dead. Shortly after I got back, lightening blew up a transformer leaving us without power for a day. When I finally got power again, I found the surge had killed the phone line and that took another day. By the time you read this, I should have a new ROM for the hard disk which I hope will eliminate the occasional problem I have with the BBS.

If you've called other boards, you'll know that the ACE BBS is extremely unusual in that there is no requirement whatsoever to upload files. As a service to you, ACE is paying for me to use PC Pursuit to exchange files with other boards — that leaves you free to download to your heart's content with no concern about not uploading! Besides, we have one of the largest collections of public domain files in the country. You might be hard pressed to find a file we don't already have!

ST files tend to be quite large even when squeezed. In order to keep our ST guests from tying up the board, I'm more stringent about requiring you to be an ACE member to download the lengthier ST files. If you have ACE membership, you'll find several hundred ST files just waiting for you to download.

— SYSOP, Ralph Walden

Atari Computer Enthusiasts

The ACE Newsletter is the first Atari user group newsletter to be published anywhere in the world. We began publishing in April, 1981.

A.C.E. is an independent, non-profit and tax exempt computer club and user group with no connection to Atari Corp. We are a group interested in educating members in the use of Atari computers and publishing the latest News, Reviews and Rumors.

All our articles, reviews and programs come from you, our members.

Our membership is world wide. Membership fees include the A.C.E. newsletter (10 issues per year) and high access in the A.C.E. Bulletin Board System. Dues are $14 a year for U.S., and $24 a year overseas airmail.

Subscription Dept.: 3662 Vine Maple Dr., Eugene, OR 97405

President — Dick Barkley, 2907 Wingate, Eugene, OR 97405
(503) 344-5843

Vice President — Larry Gold, 1927 McLean Blvd., Eugene, OR 97405
(503) 686-1490

8-Bit Librarian — Chuck & Jody Ross, 2222 Ironwood, Eugene, OR 97401
(503) 343-5545

ST Librarian — Jim Bumpas, 4405 Dillard Road, Eugene, OR 97405
(503) 484-4746

Editors — Mike Dunn, 3662 Vine Maple Dr., Eugene, OR 97405
(503) 344-6193

Jim Bumpas, 4405 Dillard Road, Eugene, OR 97405
(503) 484-4746

Larry Gold, 1927 McLean Blvd., Eugene, OR 97405
(503) 686-1490

E.R.A.C.E (Education SIG Editor) — Nora Young, 105 Hansen Lane
Eugene, OR 97404 / (503) 688-1458

Send 50c stamps or coin ($1 overseas) to the Librarian for updated ACE Library lists.

A.C.E. Bulletin Board
(503) 343-4352

On line 24 hours a day, except for servicing and updating. Consists of a 130 XE upgraded to 320k by Charlie Andrews, 2 double-density disk drives, and a 10-meg hard disk from Supra Corp, an Epson MX80 printer, a 2400 US Robotics Courier modem, running the PD/M Bulletin Board software.

Best of ACE books

Volume 1 contains bound issues of the ACE Newsletter from the first issue, Oct. 1981 to June, 1982.

Volume 2 contains July 1982 through June 1983.

Only $12 each ($2 extra for airmail). Available only from George Suetsugu, 45-602 Apuapu St., Kanoehe, HI 96744.

SortFinder Index

A composite index of Atari related articles in four popular computer periodicals, including the ACE Newsletter. Volume 1 covers April, 1981 through June, 1983. Volume 2 covers July, 1983 through December, 1985. Only $6 per printed copy or $11 per disk copy for ACE members. Available from: Jim Carr, 2660 S.W. DeArmond, Corvallis, OR 97333.

ATARI
COMPUTER
ENTHUSIASTS
3662 Vine Maple Dr. Eugene OR 97405

RENEW
NOW
==>

IF DATE EARLIER

THAN NOV 86